Claims

1.   A method of inter-process communication between at least two application processes on one computer, comprising the steps of:

5

a first process of a first application determining a name of a first file in a file system of the computer, the name of the first file being associated with a second application, the first file containing information for the first process to connect to a second process of the second application for inter-process

10   communication;

the first process initiating a first connection to the second process using the information contained in the first file;

the first process communicating with the second process using the first connection if the first connection is successfully established; and

15   the first process starting a third process of the second application if the first process fails to establish a connection with the second process.

2. The method of Claim 1, further comprising:

the first process initiating a second connection to the third process using the

20   information in the first file, in response to the third process informing the first process that the third process is ready for a connection.

3. The method of Claim 2, wherein the third process is started in a server mode without a user interface.

25

4.   The method of Claim 1, wherein the first process fails to establish a connection with the second process because the second process is not running.

5.   The method of Claim 1, wherein the first file being missing from the file system indicates that the second process is not running.

6.   The method of Claim 1, further comprising:

when the first process is started, the first process determining if a fourth process of the first application is running;

the first process requesting the fourth process to perform a task for the first process if the fourth process is running; and

the first process exiting after requesting the fourth process to perform the task for the first process.

7.   The method of Claim 6, wherein the first process determining if the fourth process of the first application is running comprises:

the first process of the first application determining a name of a second file in the file system of the computer, the name of the second file being associated with the first application.

8.   The method of Claim 7, wherein the second file being missing from the file system indicates that the fourth process of the first application is not running.

9.   The method of Claim 8, wherein the second file contains information for the first process to connect to a fourth process for inter-process communication;

failure in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is not running; and

success in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is running.

10. The method of Claim 1, wherein the first process communicates with the second process using the first connection through an Application Program Interface (API).

11. The method of Claim 10, wherein the Application Program Interface (API) is platform independent.

12. The method of Claim 1, further comprising:

when the second process is started, the second process determining if a fourth process of the second application is running;

the second process requesting the fourth process to perform a task for the second process if the fourth process is running; and

the second process exiting after requesting the fourth process to perform the task for the second process.

13. An apparatus for inter-process communication between at least two application processes on one computer, comprising the steps of:

a module for a first process of a first application determining a name of a first file in a file system of the computer, the name of the first file being

associated with a second application, the first file containing information for the first process to connect to a second process of the second application for inter-process communication;

a module for the first process initiating a first connection to the second process using the information contained in the first file;

a module for the first process communicating with the second process using the first connection if the first connection is successfully established; and

a module for the first process starting a third process of the second application if the first process fails to establish a connection with the second process.

14. The apparatus of Claim 13, further comprising:

a module for the first process initiating a second connection to the third process using the information in the first file, in response to the third process informing the first process that the third process is ready for a connection.

15. The apparatus of Claim 13, wherein the third process is started in a server mode without a user interface.

16. The apparatus of Claim 13, wherein the first process fails to establish a connection with the second process because the second process is not running.

17. The apparatus of Claim 13, wherein the first file being missing from the file system indicates that the second process is not running.

18.   The apparatus of Claim 13, further comprising:

a module for the first process determining if a fourth process of the first application is running, when, the first process is started;

a module for the first process requesting the fourth process to perform a task for the first process if the fourth process is running; and

a module for the first process exiting after requesting the fourth process to perform the task for the first process.

19.   The apparatus of Claim 18, wherein the module for the first process determining if the fourth process of the first application is running comprises:

a module for the first process of the first application determining a name of a second file in the file system of the computer, the name of the second file being associated with the first application.

20.   The apparatus of Claim 19, wherein the second file being missing from the file system indicates that the fourth process of the first application is not running.

21.   The apparatus of Claim 20, wherein the second file contains information for the first process to connect to a fourth process for inter-process communication; failure in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is not running; and success in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is running.

22. The apparatus of Claim 13, wherein the module for the first process communicates with the module for the second process using the first connection through an Application Program Interface (API).

5   23. The apparatus of Claim 22, wherein the module for the Application Program Interface (API) is platform independent.

24. The apparatus of Claim 13, further comprising:

a module for the second process determining if a fourth process of the
10   second application is running, when the second process is started;

a module for the second process requesting the fourth process to perform a task for the second process if the fourth process is running; and

a module for the second process exiting after requesting the fourth process to perform the task for the second process.

15

25. A program storage medium readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method of transmitting electronic messages in a computer environment, the method comprising the steps of:

20   a first process of a first application determining a name of a first file in a file system of the computer, the name of the first file being associated with a second application, the first file containing information for the first process to connect to a second process of the second application for inter-process communication; and

25   the first process initiating a first connection to the second process using the information contained in the first file;

the first process communicating with the second process using the first connection if the first connection is successfully established; and

the first process starting a third process of the second application if the first process fails to establish a connection with the second process.

5

26. The medium of Claim 25, wherein the method further comprises the step of:

the first process initiating a second connection to the third process using the information in the first file, in response to the third process informing the first process that the third process is ready for a connection.

10

27. The medium of Claim 26, wherein the third process is started in a server mode without a user interface.

28. The medium of Claim 25, wherein the first process fails to establish a connection with the second process because the second process is not running.

15

29. The medium of Claim 25, wherein the first file being missing from the file system indicates that the second process is not running.

20

30. The medium of Claim 25, wherein the method further comprises the steps of:

when the first process is started, the first process determining if a fourth process of the first application is running;

the first process requesting the fourth process to perform a task for the first process if the fourth process is running; and

25

the first process exiting after requesting the fourth process to perform the task for the first process.

31. The medium of Claim 30, wherein the first process determining if the fourth process of the first application is running comprises:

the first process of the first application determining a name of a second file in the file system of the computer, the name of the second file being associated with the first application.

32. The medium of Claim 31, wherein the second file being missing from the file system indicates that the fourth process of the first application is not running.

33. The medium of Claim 32, wherein the second file contains information for the first process to connect to a fourth process for inter-process communication;

failure in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is not running; and

success in connecting to the fourth process using the information contained in the second file indicates that the fourth process of the first application is running.

34. The medium of Claim 25, wherein the first process communicates with the second process using the first connection through an Application Program Interface (API).

35. The medium of Claim 34, wherein the Application Program Interface (API) is platform independent.

36. The medium of Claim 25, wherein the method further comprises the steps of:

when the second process is started, the second process determining if a fourth process of the second application is running;

the second process requesting the fourth process to perform a task for the second process if the fourth process is running; and

the second process exiting after requesting the fourth process to perform the task for the second process.

INTER-PROCESS COMMUNICATION ON A COMPUTER

Abstract